UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

| APPLICATION NO. | FILING DATE | FIRST NAMED INVENTOR | ATTORNEY DOCKET NO. | CONFIRMATION NO. |
|---|---|---|---|---|
| 10/780,407 | 02/17/2004 | Ron Ben-Natan | GRD03-04 | 1295 |

58406 7590 10/18/2007

BARRY W. CHAPIN, ESQ.
CHAPIN INTELLECTUAL PROPERTY LAW, LLC
WESTBOROUGH OFFICE PARK
1700 WEST PARK DRIVE
WESTBOROUGH, MA 01581

| EXAMINER | |
|---|---|
| STEVENS, ROBERT | |

| ART UNIT | PAPER NUMBER |
|---|---|
| 2162 | |

| MAIL DATE | DELIVERY MODE |
|---|---|
| 10/18/2007 | PAPER |

**Please find below and/or attached an Office communication concerning this application or proceeding.**

The time period for reply, if any, is set in the attached communication.

| Office Action Summary | Application No. | Applicant(s) | |
|---|---|---|---|
| | 10/780,407 | BEN-NATAN, RON | |
| | Examiner | Art Unit | |
| | Robert Stevens | 2162 | |

*-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --*

**Period for Reply**

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

**Status**

1)☒ Responsive to communication(s) filed on *31 July 2007*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

**Disposition of Claims**

4)☒ Claim(s) *1-8,11-21,23-25 and 27-43* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-8,11-21,23-25 and 27-43* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

**Application Papers**

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *19 March 2007* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

**Priority under 35 U.S.C. § 119**

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

**Attachment(s)**

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

## DETAILED ACTION

1. The Office withdraws the previous objections to the specification and drawings and the previous rejections of the claims under 35 USC §§112-2nd paragraph and 103(a), in light of the amendment. The Office maintains some (claims 21, 23-25, 27-37 and 40) withdraws some (claims 1-17) and adds some (claims 42-43) rejections of the claims under 35 USC §§101, in light of the amendment. The Office also sets forth new rejections of the claims under 35 USC §§112-2nd paragraph and 103 (a), in light of the amendment.

### *Response to Arguments*

2. Applicant's arguments with respect to the claims have been considered but are moot in view of the new ground(s) of rejection.

The Office notes that the references as a whole are believed to teach the issues that Applicant raises. A new reference, Gangadharan, has been cited to address the amended subject matter. Additionally, Gangadharan is concerned with access issues at the local level, operating behind an Internet Gateway Firewall.

### *Claim Rejections - 35 USC § 101*

3. 35 U.S.C. 101 reads as follows:

> Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

4. **Claims 21, 23-25, 27-37, 40 and 42-43 are rejected under 35 U.S.C. 101** because the claimed invention is directed to non-statutory subject matter.

To be statutory, a claimed computer-related process must either: (A) result in a physical transformation outside the computer for which a practical application is either disclosed in the specification or would have been known to a skilled artisan, or (B) be limited to a practical application with useful, concrete and tangible result.

A practical application can be either physical transformation or a useful, concrete and tangible result.

**Independent claim 21 and 42** appear to be directed to software per se (i.e., a software system or apparatus). Each of these claims encompasses a mere collection of software modules and thus lacks the necessary physical articles or objects (e.g., hardware elements) to constitute a machine or a manufacture within the meaning of 35 USC §101. Each claim is clearly not a series of steps or acts to be a process nor is it a combination of chemical compounds to be a composition of matter. As such, each fails to fall within a statutory category. Each claim is, at best, functional descriptive material *per se*.

**Claims 23-25 and 27-36** depend upon claim 21, and do not correct the deficiencies of that claim. These claims are likewise rejected.

**Claim 43** depends upon claim 42, and does not correct the deficiencies of that claim. This claim is likewise rejected.

**Independent claim 40** appears to be directed to a signal. This claim is not patent eligible because it lacks the necessary physical articles or objects to constitute a machine or a manufacture within the meaning of 35 USC 101. This claim is clearly not a series of steps or acts to be a process nor is it a combination of chemical compounds to be a composition of matter. As such, it fails to fall within a statutory category. The claim is, at best, functional descriptive material *per se*.

*Claim Rejections - 35 USC § 112*

5. **Claims 1-8, 11-21, 23-25 and 27-43 are rejected under 35 U.S.C. 112, second paragraph**, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention. These claims are vague and ambiguous, and thus, their scope is indeterminable.

**Regarding independent claim 21**: It is unclear what the subject matter of this claim is. The preamble is directed to an agent comprising a computer readable medium. An agent is software entity, but a computer readable medium appears to be a hardware entity (i.e., a memory,

according to the specification at page 7 line 9). It is unclear how a software entity can be comprised of hardware.

**Regarding independent claim 40**: It is unclear what the subject matter of this claim is. The preamble is directed to signal having code encoded on a computer readable medium. A signal is physical phenomenon, but a computer readable medium appears to be a hardware entity (i.e., a memory, according to the specification at page 7 line 9). It is unclear how a signal can be comprised of hardware.

**Regarding independent claim 42**: It is unclear what the subject matter of this claim is. The preamble appears to be reciting "instructions" subject matter. However, the terminology "processor based" is not clear (e.g., for use by a processor?).

**Regarding independent claims 1, 18, 21, 38, 39, 41 and 42**: The terminology "processor based instructions" is not clear (e.g., for use by a processor?).

**Claims 2-8 and 11-17** depend upon claim 1, and do not correct the deficiencies of that claim. These claims are likewise rejected.

**Claims 19 and 20** depend upon claim 18, and do not correct the deficiencies of that claim. These claims are likewise rejected.

**Claims 23-25 and 27-36** depend upon claim 21, and do not correct the deficiencies of that claim. These claims are likewise rejected.

**Claim 43** depends upon claim 42, and does not correct the deficiencies of that claim. This claim is likewise rejected.

### *Claim Rejections - 35 USC § 103*

6. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

1. **Claims 1-8, 11-21, 23-25 and 27-43 are rejected under 35 U.S.C. 103(a)** as being unpatentable over Krack et al. (US Patent No. 6,941,369, filed Jul. 20, 2000 and issued Sep. 6, 2005, hereafter referred to as "Krack") in view of Jai Sundar Balasubramaniyan et al.,("An Architecture for Intrusion Detection Using Autonomous Agents", 14th Annual Computer Security Applications Conf. Proc., Phoenix, AZ, Dec. 7-11, 1998, pp. 13-24, hereafter referred to as "Balasubramaniyan") and Muralidaran Gangadharan et al. ("Intranet Security with Micro-Firewalls and Mobile Agents for Proactive Intrusion Response", IEEE Int'l Conf. on Computer Networks and Mobile Computing, Beijing, China, Oct. 16-19, 2001, pp. 325-332, hereafter referred to as "Gangadharan").

**Regarding independent claim 1**: Krack discloses ***An encoded set of processor based instructions on a computer readable medium operable to perform a method of monitoring access to a protected database resource*** (See Krack Figure 3B, especially #33 and 34, showing an access control manager for a database.) ***comprising: identifying an attempt to access the database resource, the access attempt being local and directed to an access gateway of the database resource;*** (See Krack Figure 3B, especially #30, in context of column 6 lines 3-7.) ***intercepting the identified attempt to access the database resource, intercepting occurring in a prioritized manner with respect to receipt of the access attempt by the access gateway,*** (See Krack column 6 lines 29-33 discuss the interception of a database access request and column 7 lines 47-57, discussing the processing of multiple requests, it having been implicit that a prioritization scheme was necessary to handle simultaneous requests.) ***and transmitting, in a nondestructive manner, the intercepted access attempt, the nondestructive manner operable to preserve the intercepted access attempt for successive receipt by the access gateway.*** (See Krack column 6 lines 3-15, discussing the spawning of a daemon process and the nondestructive use of a unique cookie identifying the accessing party.)

Although Krack discloses the use of daemon processes, Krack does not explicitly disclose the use of agents. Balasubramaniyan, though, discloses: **the use of agents** (See Balasubramaniyan page 6 "2.2.1 Agents"; Figures 1 and 2 of page 17, showing the connection paths agents, monitors and users; and the 2nd-4th paragraphs under section "2.4.2 Audit Router" of page 19, discussing agent use in an intrusion detection environment.)

It would have been obvious to one of ordinary skill in the art at the time of the invention to apply the teachings of Balasubramaniyan for the benefit of Krack, because to do so allowed a system designer to add functionality to or remove functionality from an existing intrusion detection system without altering other system components, as taught by Balasubramaniyan on page 14 section 1.4.1 entitled "How the Use of Autonomous Agents Can Improve the Characteristics of an IDS". These references were all applicable to the same field of endeavor, i.e., access control of network-based systems.

Krack does not disclose the remaining limitations as explicitly recited. Gangadharan, though, discloses: **identifying a plurality of access paths to the protected database resource;** (See Gangadharan Fig. 1 in context of page 326 the 3rd paragraph under "2. Distributed Micro Firewalls" teaching a distributed environment, including Local Area Networks.) **intercepting further comprising: determining an IPC mechanism to be employed by a local client for accessing the DB resource;** (See Gangadharan page 326 in the 4th paragraph under "2. Distributed Micro Firewalls" teaching the use of TCP/IP.) **identifying a common access point for the access paths to the protected resource, access attempts occurring via the identified access point for the identified access paths;** (See Gangadharan Fig. 1 showing a policy manager access point.) **establishing an IPC intercept from the common access point employed by database clients for accessing the DB resource; and** (See Gangadharan Fig. 5 and page 331 in the 2nd and 3rd paragraphs under "Proactive Intrusion Response" teaching intrusion detection for TCP/IP-based systems.) **receiving the access attempt at the local agent via the IPC intercept prior to receipt of the access attempt by the access gateway;** (See

Gangadharan page 331 in the 2nd and 3rd paragraphs under "Proactive Intrusion Response" discussing the bypass of a gateway firewall and reception by a micro firewall software module.)

It would have been obvious to one of ordinary skill in the art at the time of the invention to apply the teachings of Gangadharan for the benefit of Krack in view of Balasubramaniyan, because to do so allowed a system designer to implement a distributed intrusion detection system to achieve proactive intrusion responses with dynamic policy changes, as taught by Gangadharan on page 14 section 1.4.1 entitled "How the Use of Autonomous Agents Can Improve the Characteristics of an IDS". These references were all applicable to the same field of endeavor, i.e., access control of network-based systems.

**Regarding claim 2**: Krack teaches database access and communication with a local security device. (See Krack column 9 lines 34-40, discussing access using a proprietary database call, and Figure 3B, especially #33 and 34, showing an access control manager for a database.)

Although Krack discloses the use of daemon processes, Krack does not explicitly disclose the use of agents. Balasubramaniyan, though, discloses the use of agents (See Balasubramaniyan page 6 "2.2.1 Agents"; Figures 1 and 2 of page 17, showing the connection paths agents, monitors and users; and the 2nd-4th paragraphs under section "2.4.2 Audit Router" of page 19, discussing agent use in an intrusion detection environment.)

**Regarding claim 3**: Krack teaches database access, prioritized requests and non-destructive reading of those requests. (See Krack Figure 3B in the context of column 9 lines 34-40, teaching access using a proprietary database call; column 7 lines 47-57, discussing the processing of multiple requests, it having been implicit that a prioritization scheme was necessary to handle simultaneous requests; and, column 6 lines 3-15, discussing the spawning of a daemon process and the nondestructive use of a unique cookie identifying the accessing party.)

**Regarding claim 4**: Krack discloses establishing an IPC connection prior to receiving the IPC communication. (See Krack column 8 lines 37-38, discussing the establishment of a socket-based connection to the ACM, it having been implicit that the communications path was established before sending data via that path.)

**Regarding claim 5**: Krack discloses establishing connection aggregating access attempts. (See Krack column 4 lines 55-91, discussing the handling of multiple communication requests.)

**Regarding claim 6**: Krack discloses rerouting access attempts. (See Krack column 6 lines 3-11, discussing the spawning of daemon processes to process access attempts.)

**Regarding claim 7**: Krack discloses reception of access attempts for a DB server locally and remotely via a common appliance. (See Krack Figure 3B showing a gateway device #24a and and application cgi process #35 interfacing to an access control manager #33.)

**Regarding claim 8**: Krack discloses event processing, instruction registers, DB instructions and transmission to a security device. (See Krack column 6 lines 3-11 in the context of column 7 lines 9-17, teaching forking a process upon a request event sent via a message data structure. See column 9 lines 34-40, teaching access using a proprietary database call. See Figure 3B, showing a connection to an access control manager device #33.)

**Regarding claims 11-12**: Krack discloses interprocess communications. (See Krack column 6 lines 3-15, discussing the spawning of a daemon process.)

**Regarding claim 13**: Krack discloses access interception, transmission to a security device and little effect on a host. (See Krack Figure 3B in the context of column 9 lines 34-40, teaching access using a proprietary database call. See also column 4 lines 49-54, discussing the system architecture's cost advantage over the prior art.)

**Regarding claims 14-16**: Krack discloses blocking an access request, then unblocking depending on a security decision, transmission of a decision, and logging. (See Krack Figure 3B, especially #33 and 34, showing an access control manager for a database, and column 4 lines 58-61, disclosing the use of a firewall. See also column 11 line 30, teaching logging options.)

**Regarding claim 17**: Krack discloses event processing for an IPC mechanism. (See Krack column 6 lines 3-15, discussing event processing for a request, which triggers the establishment of an IPC mechanism in the form of a socket.)

Although Krack discloses the use of daemon processes, Krack does not explicitly disclose the use of agents. Balasubramaniyan, though, discloses the use of agents. (See Balasubramaniyan page 6 "2.2.1 Agents"; Figures 1 and 2 of page 17, showing the connection paths agents, monitors and users; and the 2nd-4th paragraphs under section "2.4.2 Audit Router" of page 19, discussing agent use in an intrusion detection environment.)

It would have been obvious to one of ordinary skill in the art at the time of the invention to apply the teachings of Balasubramaniyan for the benefit of Krack, because to do so allowed a system designer to add functionality to or remove functionality from an existing intrusion detection system without altering other system components, as taught by Balasubramaniyan on page 14 section 1.4.1 entitled "How the Use of Autonomous Agents Can Improve the Characteristics of an IDS". These references were all applicable to the same field of endeavor, i.e., access control of network-based systems.

**Regarding independent claim 18**: Krack discloses ***An encoded set of processor based instructions on a computer readable medium for controlling local access to a database*** (See Krack Figure 3B, especially #33 and 34, showing an access control manager for a database.) ***comprising: identifying a local access gateway to the database, the access gateway being a common access point into the database;*** (See Krack Figure 3B, especially #33 and 34, showing

an access control manager for a database.) ***establishing an interception wrapper between a local client and the access gateway;*** (See Krack column 7 lines 9-12, discussing packetizing the message.) ***intercepting, via the interception wrapper, an access attempt from a local client prior to receipt of the access attempt by the access gateway, the access attempt indicative of a pending DB instruction in an IPC buffer identifying a local event object corresponding to the access attempt;*** (See Krack column 6 lines 3-11, discussing IPC processing.) ***indexing a notification list corresponding to the identified local event object;*** (See Krack column 6 lines 23-26, discussing the user database.) ***traversing the indexed notification list, the notification list including entries of notifications to be performed upon occurrence of the event;*** (See Krack column 6 lines 23-26, discussing reading from the user database.) ***reading a traversed entry;*** (See Krack column 6 lines 23-26, discussing reading from the user database.) ***retrieving, in response to the notification, the DB instruction from the IPC buffer;*** (See Krack column 9 lines 34-40, discussing access using a proprietary database call, and Figure 3B, especially #33 and 34, showing an access control manager for a database.) ***transmitting the retrieved DB instruction from the IPC buffer to a data security device operable to analyze the propriety of the DB instruction;*** (See Krack column 9 lines 34-40, teaching access using a proprietary database call. See Figure 3B, showing a connection to an access control manager device #33, and column 6 lines 11-21, discussing authentication.) ***reading a successive traversed entry corresponding to the access gateway, the entry indicative of the location of the access gateway;*** (See Krack column 7 lines 47-56, discussing multiple requests.) ***and notifying the access gateway of the IPC event occurrence using the read location of the access gateway.*** (See Krack column 6 lines 17-26, discussing user validation.)

Although Krack discloses the use of daemon processes, Krack does not explicitly disclose the use of agents. Balasubramaniyan, though, discloses: ***reading a traversed entry corresponding to the local agent, the entry indicative of the location of the local agent; notifying the local agent using the read location of the local agent*** (See Balasubramaniyan page 6 "2.2.1 Agents"; Figures 1 and 2 of page 17, showing the connection paths agents, monitors and users; and the 2nd-4th paragraphs under section "2.4.2 Audit Router" of page 19, discussing agent use in an intrusion detection environment.)

It would have been obvious to one of ordinary skill in the art at the time of the invention to apply the teachings of Balasubramaniyan for the benefit of Krack, because to do so allowed a system designer to add functionality to or remove functionality from an existing intrusion detection system without altering other system components, as taught by Balasubramaniyan on page 14 section 1.4.1 entitled "How the Use of Autonomous Agents Can Improve the Characteristics of an IDS". These references were all applicable to the same field of endeavor, i.e., access control of network-based systems.

Krack does not disclose the remaining limitations as explicitly recited. Gangadharan, though, discloses: **establishing the interception wrapper further comprising: identifying at least one interprocess communication operation, each of the identified IPC operation corresponding to an event, the event derived from a database (DB) instruction;** (See Gangadharan page 326 in the 4th paragraph under "2. Distributed Micro Firewalls" teaching the use of TCP/IP.) **instantiating a local event object corresponding to the event, the local event**

**object having a notification list indicative of notifications of an object to be made upon an occurrence of the event; and** (See Gangadharan Fig. 5 and page 331 in the 2nd and 3rd paragraphs under "Proactive Intrusion Response" teaching intrusion detection for TCP/IP-based systems.) **storing, in a first position in the notification list, an indication of the local agent, the first position operable to provide the first notification upon an occurrence of the event, prior to other notifications in the notification list;** (See Gangadharan Fig. 5 and page 331 in the 2nd and 3rd paragraphs under "Proactive Intrusion Response" teaching intrusion detection for TCP/IP-based systems.)

It would have been obvious to one of ordinary skill in the art at the time of the invention to apply the teachings of Gangadharan for the benefit of Krack in view of Balasubramaniyan, because to do so allowed a system designer to implement a distributed intrusion detection system to achieve proactive intrusion responses with dynamic policy changes, as taught by Gangadharan on page 14 section 1.4.1 entitled "How the Use of Autonomous Agents Can Improve the Characteristics of an IDS". These references were all applicable to the same field of endeavor, i.e., access control of network-based systems.

**Regarding claim 19**: Krack discloses event processing, an IPC mechanism and storage. (See Krack column 6 lines 3-11 in the context of column 7 lines 9-17, teaching forking a process upon a request event sent via a message data structure. See column 9 lines 34-40, teaching access using a proprietary database call. See Figure 3B, showing a connection to an access

control manager device #33. See Krack column 6 lines 3-15, discussing the spawning of a daemon process. See also Figure 3B, #36, showing local data storage.)

**Regarding claim 20**: Krack discloses message reception, processing and transmission. (See Krack column 7 lines 9-17, discussing message processing, it having been implicit that such messages were also received and transmitted.)

**Claims 21-25 and 27-37** are directed to agents that implement the methods of claim 1-5 and 7-17, respectively, and therefore likewise rejected.

**Regarding independent claim 38**: Krack discloses ***A data security device for monitoring access to a protected database resource*** (See Krack Figure 3B, especially #33 and 34, showing an access control manager for a database.) ***comprising: a memory comprising a computer readable medium operable to store an encoded set of processor based instructions;*** (See Krack Figure 3B, it having been implicit that a memory was necessary to store the code for daemon process #32.) ***a processor operable to execute instructions in the memory;*** (See Krack Figure 3B, it having been implicit that a processor was used to run the executable daemon process #32.) ***an interface operable for interconnection with a database host, the data security device in communication with the database host,*** (See Krack Figure 3B, especially #33 and 34, showing an access control manager for a database.) ***identify an attempt to access the database resource, the access attempt being local and directed to an access gateway of the database***

***resource;*** (See Krack Figure 3B, especially #30, in context of column 6 lines 3-7.) ***intercept the identified attempt to access the database resource, intercepting occurring in a prioritized manner with respect to receipt of the access attempt by the access gateway,*** (See Krack column 6 lines 29-33 discuss the interception of a database access request and column 7 lines 47-57, discussing the processing of multiple requests, it having been implicit that a prioritization scheme was necessary to handle simultaneous requests.) ***and transmit, in a nondestructive manner, the intercepted access attempt, the nondestructive manner operable to preserve the intercepted access attempt for successive receipt by the access gateway.*** (See Krack column 6 lines 3-15, discussing the spawning of a daemon process and the nondestructive use of a unique cookie identifying the accessing party.)

Although Krack discloses the use of daemon processes, Krack does not explicitly disclose the use of agents. Balasubramaniyan, though, discloses: **the use of agents** (See Balasubramaniyan page 6 "2.2.1 Agents"; Figures 1 and 2 of page 17, showing the connection paths agents, monitors and users; and the 2nd-4th paragraphs under section "2.4.2 Audit Router" of page 19, discussing agent use in an intrusion detection environment.)

It would have been obvious to one of ordinary skill in the art at the time of the invention to apply the teachings of Balasubramaniyan for the benefit of Krack, because to do so allowed a system designer to add functionality to or remove functionality from an existing intrusion detection system without altering other system components, as taught by Balasubramaniyan on page 14 section 1.4.1 entitled "How the Use of Autonomous Agents Can Improve the

Characteristics of an IDS". These references were all applicable to the same field of endeavor, i.e., access control of network-based systems.

Krack does not disclose the remaining limitations as explicitly recited. Gangadharan, though, discloses: **intercepting comprising: identifying at least one interprocess communication operation, each of the identified IPC operation corresponding to an event, the event derived from a database (DB) instruction;** (See Gangadharan page 326 in the 4th paragraph under "2. Distributed Micro Firewalls" teaching the use of TCP/IP.) **instantiating a local event object corresponding to the event, the local event object having a notification list indicative of notifications of an object to be made upon an occurrence of the event; and** (See Gangadharan Fig. 5 and page 331 in the 2nd and 3rd paragraphs under "Proactive Intrusion Response" teaching intrusion detection for TCP/IP-based systems.) **storing, in a first position in the notification list, an indication of the local agent, the first position operable to provide the first notification upon an occurrence of the event, prior to other notifications in the notification list;** (See Gangadharan Fig. 5 and page 331 in the 2nd and 3rd paragraphs under "Proactive Intrusion Response" teaching intrusion detection for TCP/IP-based systems.)

It would have been obvious to one of ordinary skill in the art at the time of the invention to apply the teachings of Gangadharan for the benefit of Krack in view of Balasubramaniyan, because to do so allowed a system designer to implement a distributed intrusion detection system to achieve proactive intrusion responses with dynamic policy changes, as taught by Gangadharan on page 14 section 1.4.1 entitled "How the Use of Autonomous Agents Can Improve the

Characteristics of an IDS". These references were all applicable to the same field of endeavor, i.e., access control of network-based systems.

**Independent claims 39-42** are respectively directed to a computer program product, a signal, and a device for the implementation of claim 1, and as such are likewise rejected.

**Regarding claim 43**: Krack does not disclose the remaining limitations as explicitly recited. Gangadharan, though, discloses: **intercepting comprising: identifying at least one interprocess communication operation, each of the identified IPC operation corresponding to an event, the event derived from a database (DB) instruction;** (See Gangadharan page 326 in the 4th paragraph under "2. Distributed Micro Firewalls" teaching the use of TCP/IP.) **instantiating a local event object corresponding to the event, the local event object having a notification list indicative of notifications of an object to be made upon an occurrence of the event; and** (See Gangadharan Fig. 5 and page 331 in the 2nd and 3rd paragraphs under "Proactive Intrusion Response" teaching intrusion detection for TCP/IP-based systems.) **storing, in a first position in the notification list, an indication of the local agent, the first position operable to provide the first notification upon an occurrence of the event, prior to other notifications in the notification list;** (See Gangadharan Fig. 5 and page 331 in the 2nd and 3rd paragraphs under "Proactive Intrusion Response" teaching intrusion detection for TCP/IP-based systems.)

It would have been obvious to one of ordinary skill in the art at the time of the invention to apply the teachings of Gangadharan for the benefit of Krack in view of Balasubramaniyan, because to do so allowed a system designer to implement a distributed intrusion detection system to achieve proactive intrusion responses with dynamic policy changes, as taught by Gangadharan on page 14 section 1.4.1 entitled "How the Use of Autonomous Agents Can Improve the Characteristics of an IDS". These references were all applicable to the same field of endeavor, i.e., access control of network-based systems.

## *Conclusion*

7. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

### *Non-Patent Literature.*

Miller, Sandra Kay, "The Trusted OS Makes a Comeback", Computer, vol. 34, Issue 2, Feb. 2001, pp. 16-19.

Microsoft Computer Dictionary, 5th Edition, Microsoft Press, Redmond, WA, © 2002, page 22.

### *US Patent Application Publications*

| | |
|---|---|
| Jo et al | 2005/0071650 |
| Lee | 2003/0182580 |

### *US Patents*

| | |
|---|---|
| Jade et al | 6,061,797 |
| See et al | 6,070,243 |
| Shipley | 6,119,236 |
| Shipley | 6,304,975 |

8. **THIS ACTION IS MADE FINAL**. Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

***Contact Information***

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Robert Stevens whose telephone number is (571) 272-4102. The examiner can normally be reached on M-F 6:00 - 2:30.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, John E. Breene can be reached on (571) 272-4107. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

Robert Stevens
Examiner
Art Unit 2162

October 7, 2007

JOHN BREENE
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100